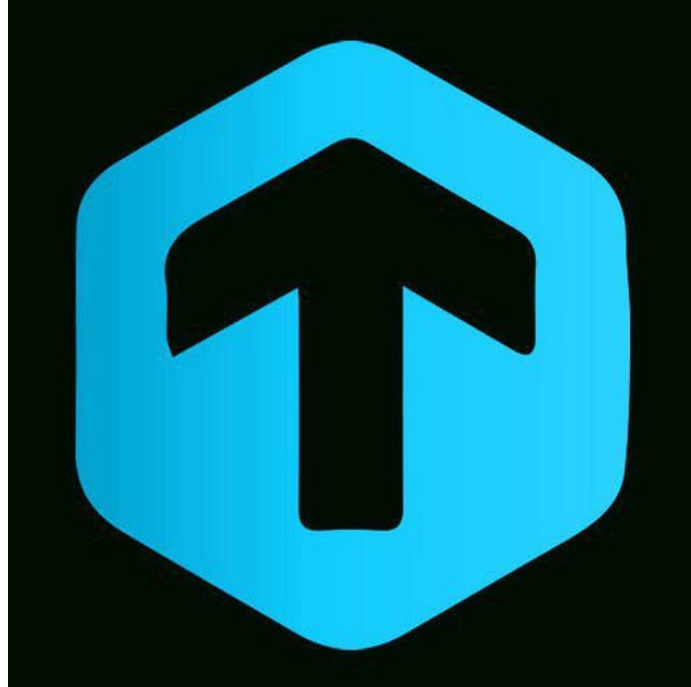




PREMIUM AUDIT

NAME OF PROJECT	Evameta
TOKEN BEP20 ADDRESS	0x02Ccb9bE58c1C13AB2830733fabef8BC39F6C0A6
WEBSITE / TELEGRAM	evameta.finance



DISCLAIMER: PLEASE READ FULL AUDIT



IMPORTANT DISCLAIMER

I.

NOT RESPONSIBLE

Our audit report is not responsible for anything if the project audited turn into a scam.

II.

NOT GUARANTEE

Analytix Audit in no way guarantees that a project will not remove liquidity, sell off team tokens, or exit scam.

III.

INFORMATION

Analytix Audit do a research and provides public information about the project in an easy-to-understand format for the common person.

IV.

AUDIT AGREE

Agreeing to an audit in no way guarantees that a team will not remove all liquidity ("Rug Pull"), remove liquidity slowly, sell off tokens, quit the project, or completely exit scam

V.

ENCOURAGE INVESTMENT

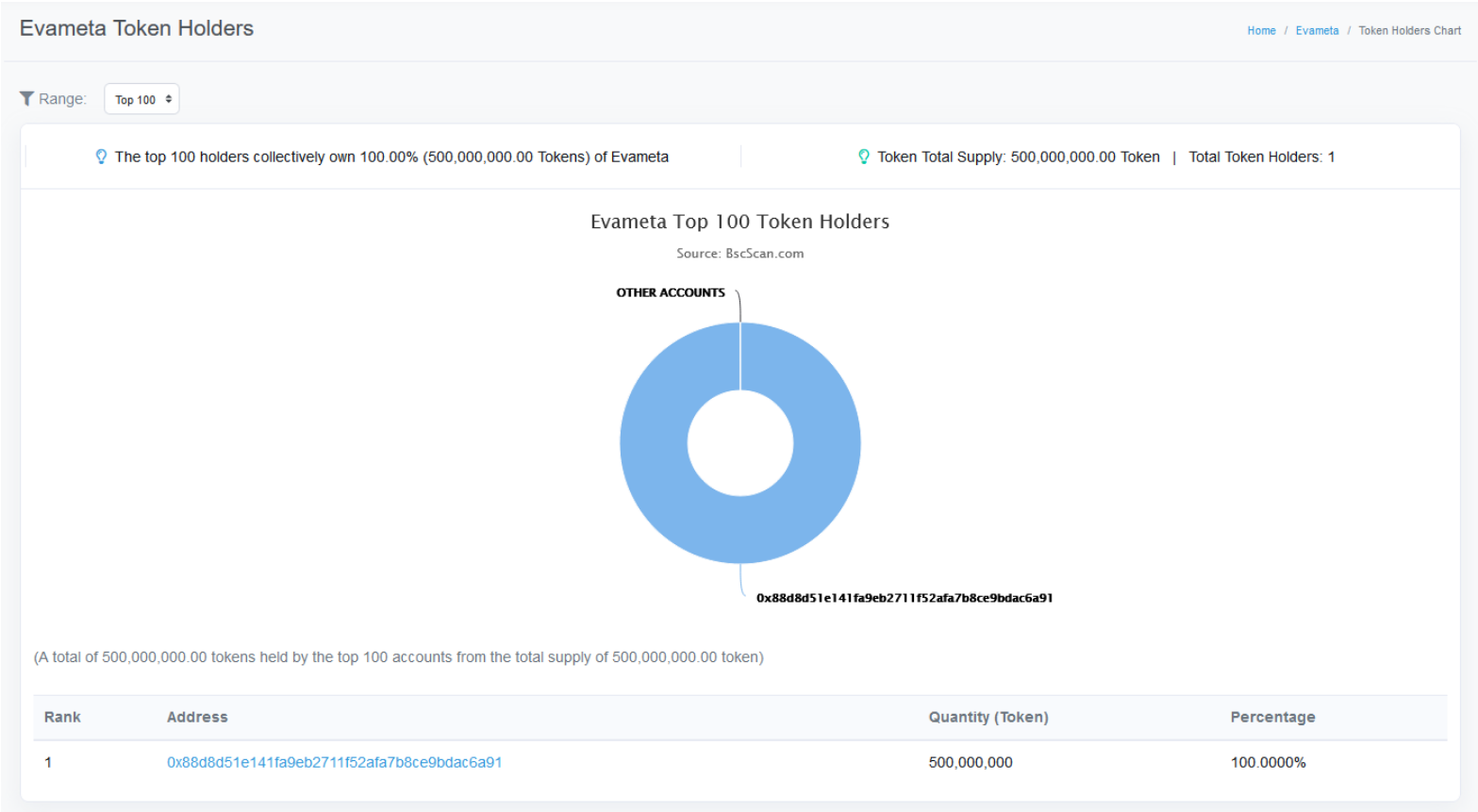
Analytix Audit in no way takes responsibility for any losses, nor does Analytix Audit encourage any speculative investments

VI.

INFORMATION

The information provided in this audit is for information purposes only and should not be considered investment advice. Analytix Audit does not endorse, recommend, support, or suggest any projects that have been audited.

TOKENOMICS - OVERVIEW



- More than 15% of the tokens are unlocked

(This comment refers to when the audit was carried out maybe team locked tokens later)

CONTRACT CODE - OVERVIEW



QUICK REVIEW

Current Fees:

Buy: 11%

Sell: 11%

Owner can change fees up to 40%

Owner can change Tx Limits.

(Minium Tx Limit: 0.005% Supply) ~ 25.000 EVAM

Owner Privileges:

```
520 * Requirements:
521 *
522 * - the calling contract must have an ETH balance of at least `value`.
523 * - the called Solidity function must be `payable`.
524 *
525 * _Available since v3.1._
526 */
527 function functionCallWithValue(
528     address target,
529     bytes memory data,
530     uint256 value
531 ) internal returns (bytes memory) {
532     return functionCallWithValue(target, data, value, "Address: low-level call with value failed");
533 }
534
535 /**
536 * @dev Same as {xref-Address-functionCallWithValue-address-bytes-uint256}[`functionCallWithValue`], but
537 * with `errorMessage` as a fallback revert reason when `target` reverts.
538 *
539 * _Available since v3.1._
540 */
541 function functionCallWithValue(
542     address target,
543     bytes memory data,
544     uint256 value,
545     string memory errorMessage
546 ) internal returns (bytes memory) {
547     require(address(this).balance >= value, "Address: insufficient balance for call");
548     require(isContract(target), "Address: call to non-contract");
549
550     (bool success, bytes memory returndata) = target.call(value: value)(data);
```

```
function rescueBEP20Tokens(address tokenAddress) external onlyOwner{
function excludeFromFees(address account, bool excluded) public onlyOwner {
function excludeMultipleAccountsFromFees(address[] calldata accounts, bool excluded) public onlyOwner {
function excludeFromDividends(address account, bool value) external onlyOwner{
function setmarketingWallet(address newmarketingWallet) external onlyOwner{
function setMaxWalletBalance(uint256 amount) external onlyOwner{
function setMaxTxAmount(uint256 amount) external onlyOwner{
function setSwapTokensAtAmount(uint256 amount) external onlyOwner{
function setTaxes(uint256 _rewards, uint256 _liquidity, uint256 _marketing, uint256 _extraSell) external
    onlyOwner{
    function setSwapEnabled(bool _enabled) external onlyOwner{
    function setBot(address bot, bool value) external onlyOwner{
function setAutomatedMarketMakerPair(address newPair, bool value) external onlyOwner {
    function setGasForProcessing(uint256 newValue) external onlyOwner {
    function setClaimWait(uint256 claimWait) external onlyOwner {
        modifier onlyOwner() {
            function renounceOwnership() public virtual onlyOwner {
            function transferOwnership(address newOwner) public virtual onlyOwner {
            function distributeRewardTokenDividends(uint256 amount) public onlyOwner{
function excludeFromDividends(address account, bool value) external onlyOwner {
    function updateClaimWait(uint256 newClaimWait) external onlyOwner {
    function setBalance(address account, uint256 newBalance) external onlyOwner {
function processAccount(address account, bool automatic) public onlyOwner returns (bool) {
```

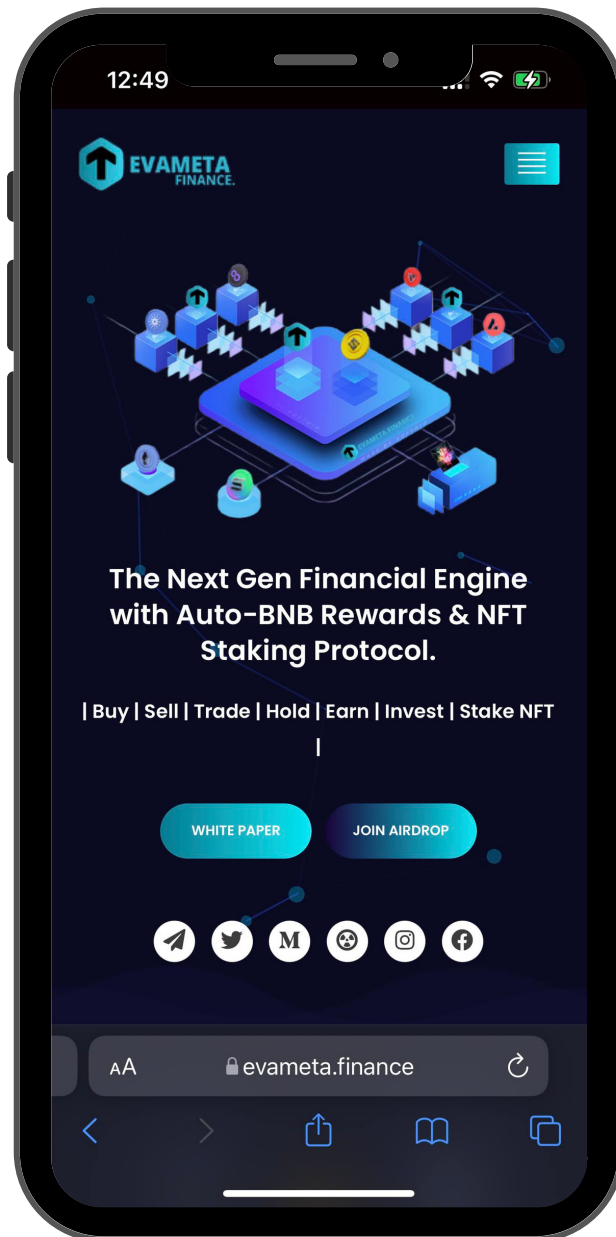
Issue Checking:

N°	Issue description.	Checking Status.
1	Compiler Errors	Passed
2	Race conditions and Reentrancy. Cross-function race conditions.	Passed
3	Possible delays in data delivery.	Passed
4	Oracle calls.	Passed
5	Front running.	Passed
6	Timestamp dependence.	Passed
7	Integer Overflow and Underflow.	Passed
8	DoS with Revert.	Passed
9	DoS with block gas limit.	Passed
10	Methods execution permissions.	Passed
11	Economy model.	Passed
12	The impact of the exchange rate on the logic.	Passed
13	Private user data leaks.	Passed
14	Malicious Event log	Passed
15	Scoping and Declarations.	Passed
16	Uninitialized storage pointers.	Passed
17	Arithmetic accuracy.	Passed
18	Design Logic.	Passed
19	Cross-function race conditions.	Passed
20	Safe Zeppelin module	Passed
21	Fallback function security.	Passed

Audit Result:

PASSED

WEBSITE - OVERVIEW



🕒 **Domain Registration: 2022-01-07**

45 days old

SSL CERTIFICATE: A

(<https://www.ssllabs.com>)

SOCIAL MEDIA - OVERVIEW



EVAMETA_FINANCE



EVAMETA_FINANCE



TAYZTZZG92



EVAMETAFINANCE.MEDIUM.COM



EVAMETA_FINANCE



EVAMETA.FINANCE